

```

restart
S := readdata( "E:/TNOs/aeV-R.txt", 3 )
F := (x, y, z) → x2 + b y2 + c z2 + 2 d x y + 2 e x z + 2 f y z + g x + h y + i z + j
eqns := map(F@op, convert(S, set))
linalgleastsqrs(eqns, {b, c, d, e, f, g, h, i, j})
{c = -1594.827695, d = -36.73945508, e = 7.008045672, f = 1734.181721, b = -6339.295891,
g = -58.86482353, h = 3616.689156, i = 715.1995081, j = 311.6372556}
subs(%, F(x, y, z))
fit := fn(%)
fit := (x, y, z) → x2 - 6339.295891 y2 - 1594.827695 z2 - 73.47891016 x y + 14.01609134 x z
+ 3468.363442 y z - 58.86482353 x + 3616.689156 y + 715.1995081 z + 311.6372556
ContourPlanePlot := proc(F::procedure, v1min::{integer, float}, v1max::{integer, float},
v2min::{integer, float}, v2max::{integer, float}, v0::{integer, float}, v3min::{integer, float}, plane::string,
Ncontours::posint, cgrid::list, contour_colors::list)
local p, f, tmp;
if plane = "xy" then
    p := plots[contourplot](F(x, y, v0), x = v1min .. v1max, y = v2min .. v2max, contours = Ncontours,
                           filled = true, coloring = contour_colors, grid = cgrid);
    f := plottools[transform]((x, y) → [x, y, v3min])
elif plane = "yz" then
    p := plots[contourplot](F(v0, y, z), y = v1min .. v1max, z = v2min .. v2max, contours = Ncontours,
                           filled = true, coloring = contour_colors, grid = cgrid);
    f := plottools[transform]((y, z) → [v3min, y, z])
else
    p := plots[contourplot](F(x, v0, z), x = v1min .. v1max, z = v2min .. v2max, contours = Ncontours,
                           filled = true, coloring = contour_colors, grid = cgrid);
    f := plottools[transform]((x, z) → [x, v3min, z])
fi;
f(p)
end
Warning, `v3min` is a lexically scoped parameter
Warning, `v3min` is a lexically scoped parameter
Warning, `v3min` is a lexically scoped parameter
VRmin := .2
VRmax := 1.2
amin := 17
amax := 50
emin := 0
emax := .7

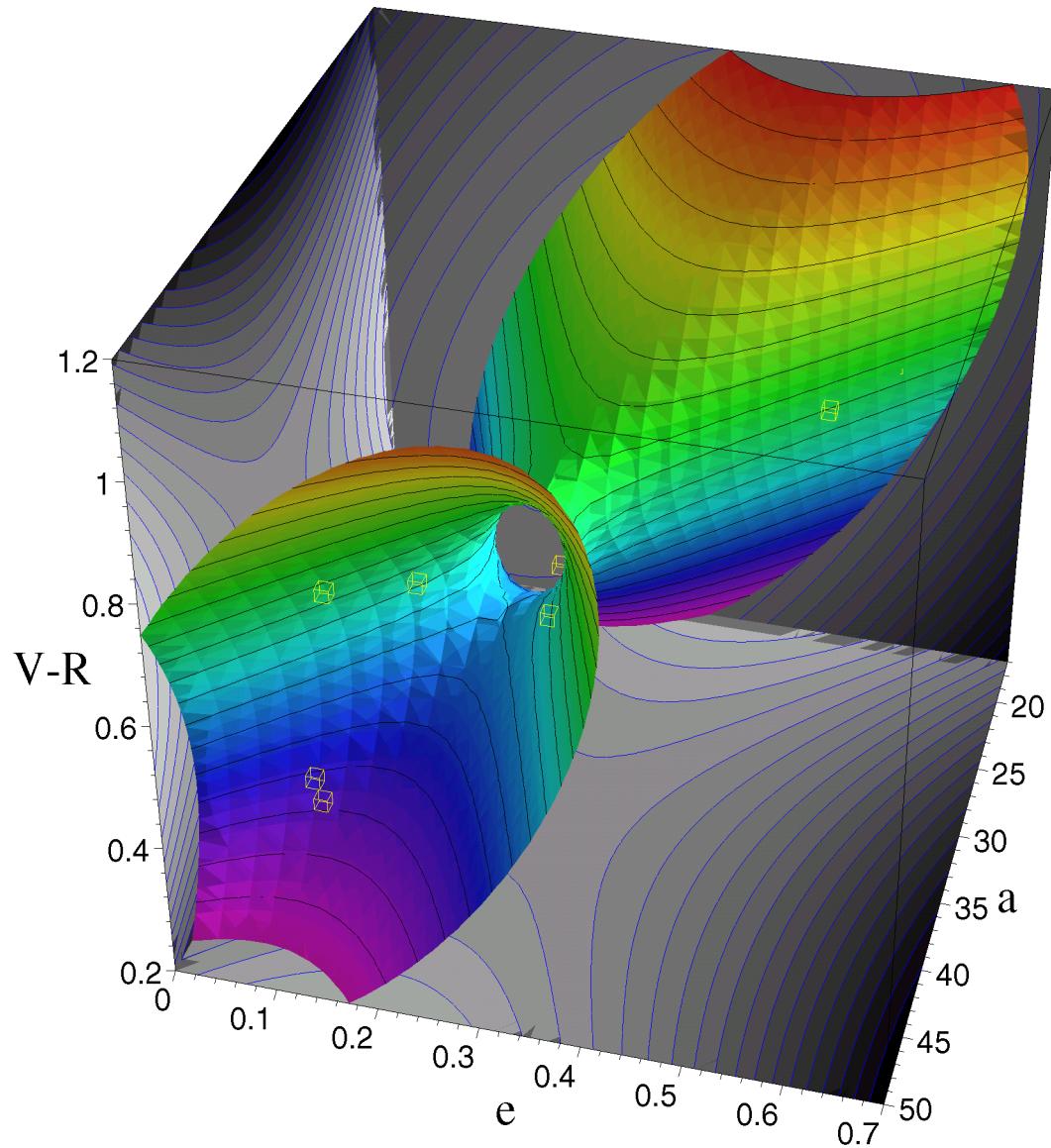
```

```

VR0 := .7
e0 := .25
a0 := 35
Ncontours := 20
contour_grid := [40, 40]
contour_colors := [black, gray]
p_surf := plots implicitplot3d(fit(x, y, z), x = amin .. amax, y = emin .. emax, z = VRmin .. VRmax,
style = patchcontour, contours = Ncontours, shading = zhue, grid = [30, 30, 30])
p_points := plots pointplot3d(S, color = yellow, symbol = box)
p_xy :=
ContourPlanePlot(fit, amin, amax, emin, emax, VR0, VRmin, "xy", Ncontours, contour_grid, contour_colors)
p_xz :=
ContourPlanePlot(fit, amin, amax, VRmin, VRmax, e0, emin, "xz", Ncontours, contour_grid, contour_colors)
p_yz :=
ContourPlanePlot(fit, emin, emax, VRmin, VRmax, a0, amin, "yz", Ncontours, contour_grid, contour_colors)
plots display3d({p_surf, p_points, p_xy, p_xz, p_yz}, projection = .9, labels = ["a", "e", "V-R"],
title = "V-R vs. (a,e)", orientation = [15, 55], view = [amin .. amax, emin .. emax, VRmin .. VRmax])

```

V-R vs. (a,e)



```

p_surf := plots[implicitplot3d](fit(x, y, z), x = amin .. amax, y = emin .. emax, z = VRmin .. VRmax,
style = patchcontour, contours = Ncontours, projection = .9, labels = ["a", "e", "V-R"], title = "V-R vs. (a,e)",
orientation = [15, 55], view = [amin .. amax, emin .. emax, VRmin .. VRmax], shading = zhue,
grid = [10, 10, 10])

```

```

p_surf := plots[implicitplot3d]

$$\left( \text{fit}(x \text{ amax}, y, z), x = \frac{\text{amin}}{\text{amax}} .. 1, y = \text{emin} .. \text{emax}, z = \text{VRmin} .. \text{VRmax}, \right.$$

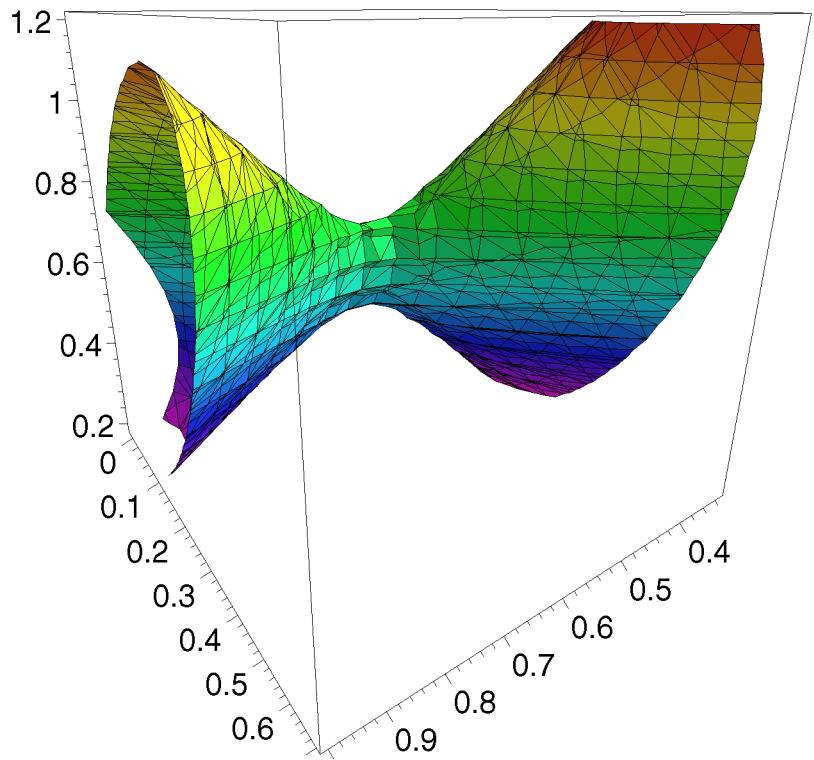

$$\left. \text{grid} = [15, 15, 15], \text{axes} = \text{box} \right)$$

```

```

p_surf

```



```
[plottoolsvrml(p_points, "E:/TNOs/aeV-R.wrl")
```

```
[ ?
```